

CLAIMS

1. A heuristics analysis tool comprising:
a persistent table, having clean data records and key records wherein there is at least one key record associated with each clean data record, said key record having at least one field of data from an associated said clean data record;
and
associated with said key records, heuristic-based routines for automatically generating said key records from each newly received data record for matching to said clean data record.
2. The tool as set forth in claim 1 wherein said clean data record is a primary clean key record of a plurality of said key records in a set .
3. The tool as set forth in claim 2 wherein said primary clean key record is a pointer to a complete data file associated with a clean data record.
4. The tool as set forth in claim 1 wherein said clean data record is a primary complete clean data file.
5. The tool as set forth in claim 1 further comprising:
at least one column recording one or more of said heuristic-based routines

1 that were involved in generating each of said key records.

2 6. The tool as set forth in claim 1 further comprising:

3 a time-stamp with each said key record in the table wherein said time-
4 stamp indicative of most recent use.

5 7. The tool as set forth in claim 1 further comprising:

6 special flags associated with said key records, said flags associated with
7 specific heuristic considerations .

8 8. The tool as set forth in claim 7 wherein a special flag is a quality factor
9 assigned to each said key record.

10 9. A data association and cleaning method comprising:

11 storing a plurality of clean data files and, associated with each of said
12 clean data files, at least one indexing record, each said indexing record
13 containing at least one field related to a respective associated clean data file
14 such that said at least one indexing record serves as a pointer to the respective
15 associated said clean data file;

16 comparing incoming data records to the indexing records for obtaining a
17 match, and assigning said input data record to the respective associated said
18 clean data file associated with a matched indexing record;

1 if not obtaining a said match, iteratively cleaning input data until at least a
2 near-match between said input data and said at least one indexing record is
3 obtained and assigning said input data record to the one of said clean data files
4 associated with a near-matched indexing record; and

5 upon a near match, adding a so-cleaned input data record as a new
6 indexing record for an associated one of said clean data files, and upon no
7 match, adding said so-cleaned input data record as a new clean data file with an
8 associated indexing record therefor.

9 10. The method as set forth in claim 9 wherein said storing is in a displayable
10 format.

11 11. The method as set forth in claim 10 further comprising:
12 at given intervals, performing a data clean-up on a stored table in said
13 displayable format.

14 12. The method as set forth in claim 9 wherein upon said adding said so-
15 cleaned input data record as a new clean data file with an associated indexing
16 record therefor, flagging said new clean data file.

17 13. The method as set forth in claim 9, said iteratively cleaning further
18 comprising:

1 upon said not recognizing a match therebetween, cleaning said input data
2 and storing a first cleaned input data set;
3 comparing the first cleaned input data to each said indexing record, and
4 upon recognizing a match therebetween, stopping said comparing
5 and retrieving the associated clean data for association with said input data,
6 upon not recognizing a match therebetween, re-cleaning said first
7 cleaned input data, discarding said first cleaned input data, and storing a
8 subsequently cleaned input data set;
9 re-comparing the subsequently cleaned input data set to said indexing
10 record; and
11 iteratively repeating said re-cleaning and re-comparing until a
12 predetermined phase of cleaning is reached without the said match therebetween
13 wherein said a last said subsequently cleaned input data is stored as a new clean
14 data file.

15 14. The method as set forth in claim 13 wherein upon said recognizing a
16 match therebetween, a new indexing record is generated for said new clean data
17 file.

18 15. A computer memory comprising:
19 computer code means for receiving input data records;
20 computer code means for comparing said input data records to a tabular

1 format set of crude keys;

2 computer code means for returning a clean key associated with one of said
3 crude keys upon a comparing match;

4 computer code means for iterative cleaning of said input data records upon
5 a no-match return and storing an iteratively-generated respective cleaned data
6 record therefrom;

7 computer code means for re-comparing said iteratively-generated
8 respective cleaned data record to said set of crude keys, searching for said
9 match return; and

10 computer code means for creating a new data file from a last said
11 iteratively-generated respective cleaned data record such that said new data file
12 is also a first respective one of said crude keys associated therewith.

13 16. The computer memory as set forth in claim 15 further comprising:

14 computer code means for generating a new crude key from an iteratively-
15 generated respective cleaned data record.

16 17. The computer memory as set forth in claim 16 wherein said computer code
17 means for generating a new crude key has heuristic routines.

18 18. The computer memory as set forth in claim 17 further comprising:

19 computer code means for displaying in said tabular format said crude keys

1 and heuristic routines employed in said generating.

2 19. The computer memory as set forth in claim 15 wherein each of said crude
3 keys has an associated pointer to obtain said associated clean key.

4 20. The computer memory as set forth in claim 17 wherein each of said crude
5 keys points to a cleanest one of a plurality of crude keys associated with a clean
6 data file.

7 21. The computer memory as set forth in claim 15 wherein said tabular array is
8 a displayable table, comprising:
9 computer code means including heuristic routines for editing said table.

10 22. The computer memory as set forth in claim 15 wherein said tabular array is
11 a displayable table comprising:
12 computer code means including heuristic routines for extending fields of
13 data fields in said table.

1 23. A method of doing business comprising:
2 storing a database of clean data files for each of a plurality of entities;
3 creating a tabulation of crude keys, each having a pointer to an associated
4 one of said clean data files;
5 receiving periodically a potentially dirty data record related to at least one
6 entity of said plurality of entities;
7 comparing said dirty data to said array;
8 assigning said dirty data to one of said clean data files; and
9 creating new clean data files from said dirty data when no pointer
10 substantially matching said dirty data is found during said comparing.